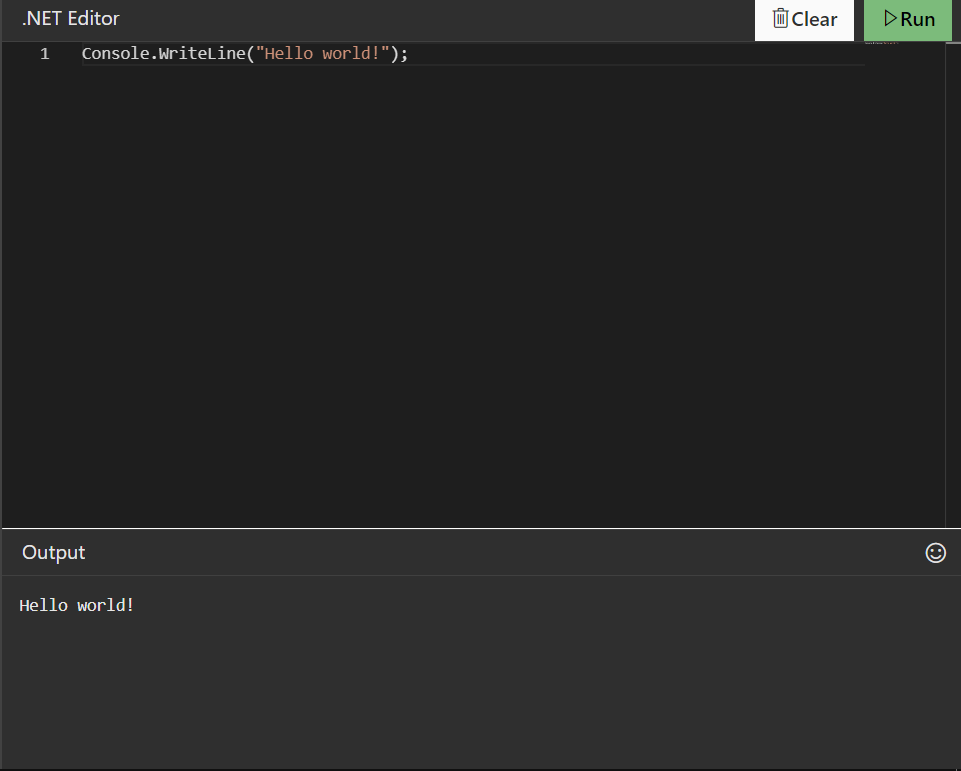
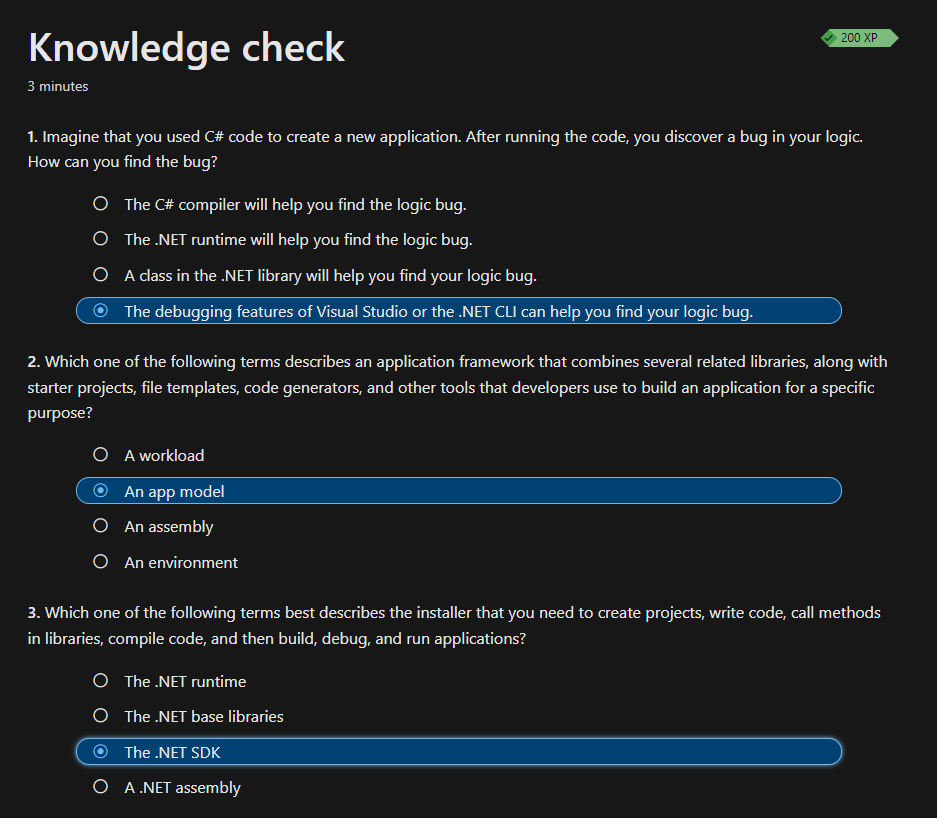
**SESSION - 1:**

**Build your first app by using Try .NET**

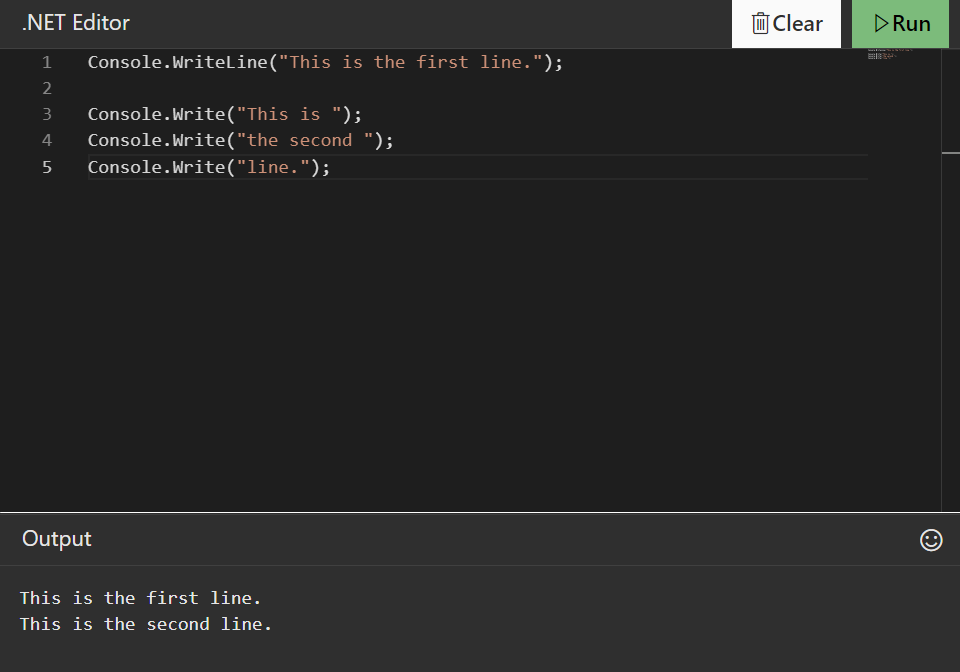


**MCQ :**

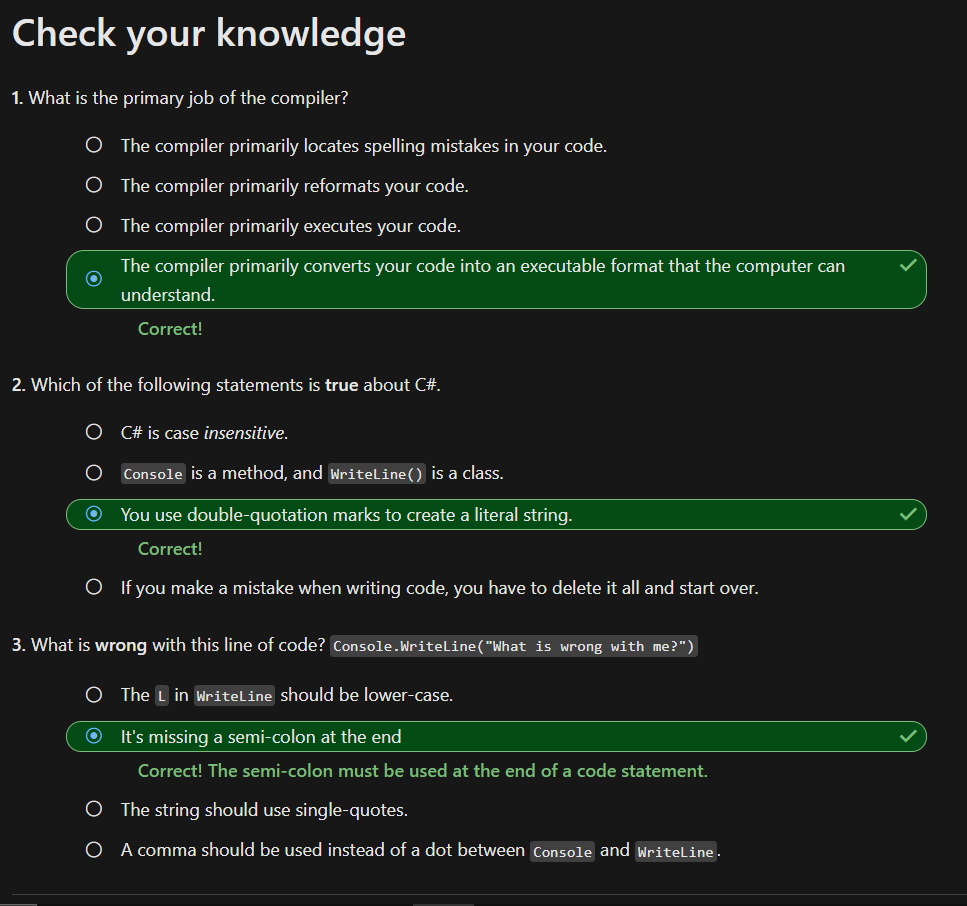


**SESSION - 2:**

**Write code to display two messages**

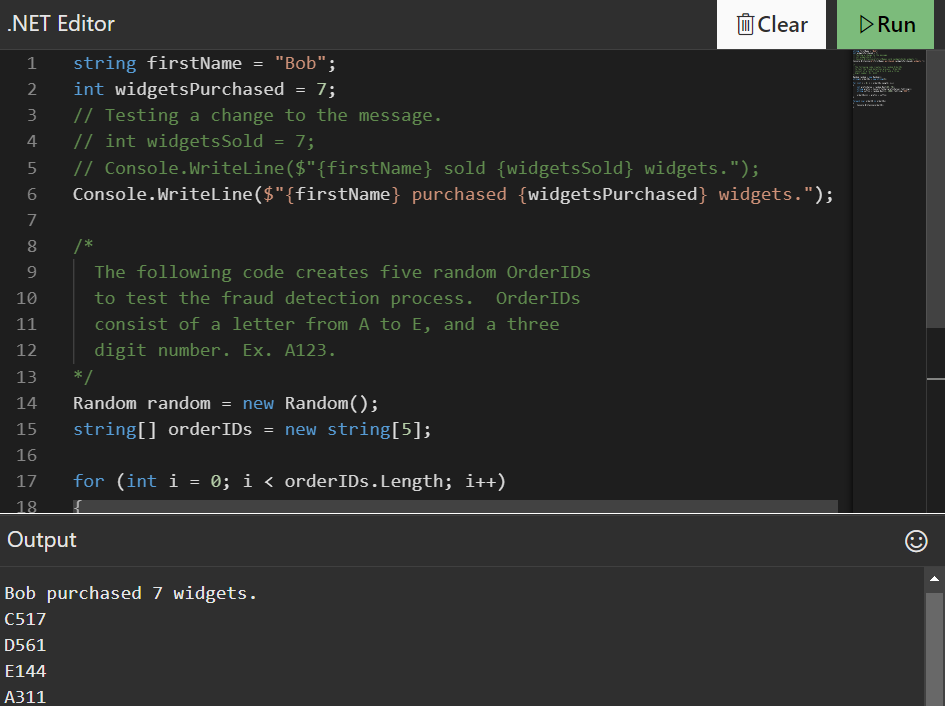


**MCQ :**

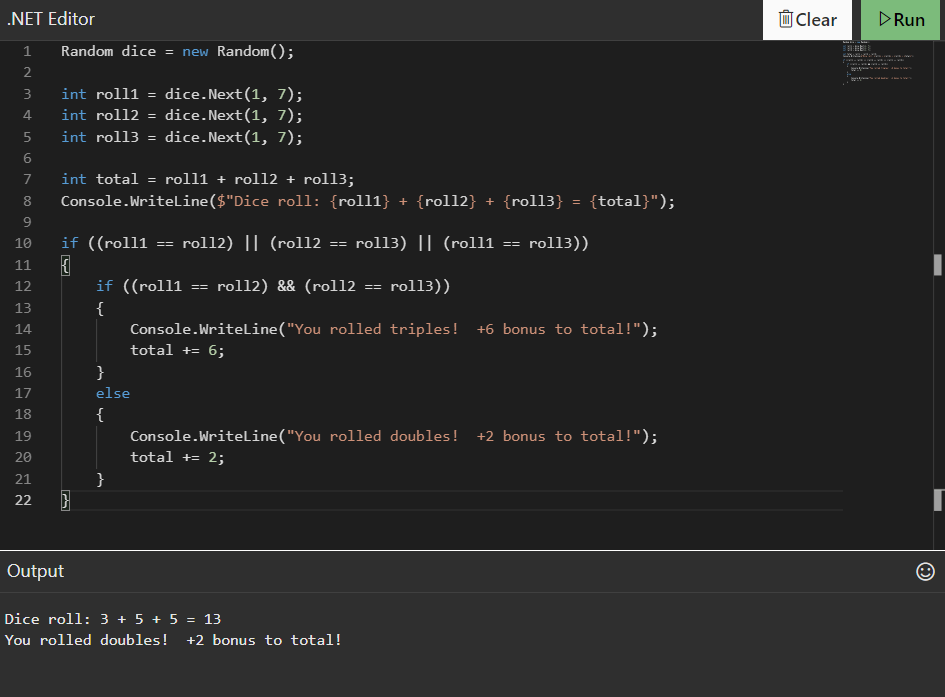


**SESSION - 3:**

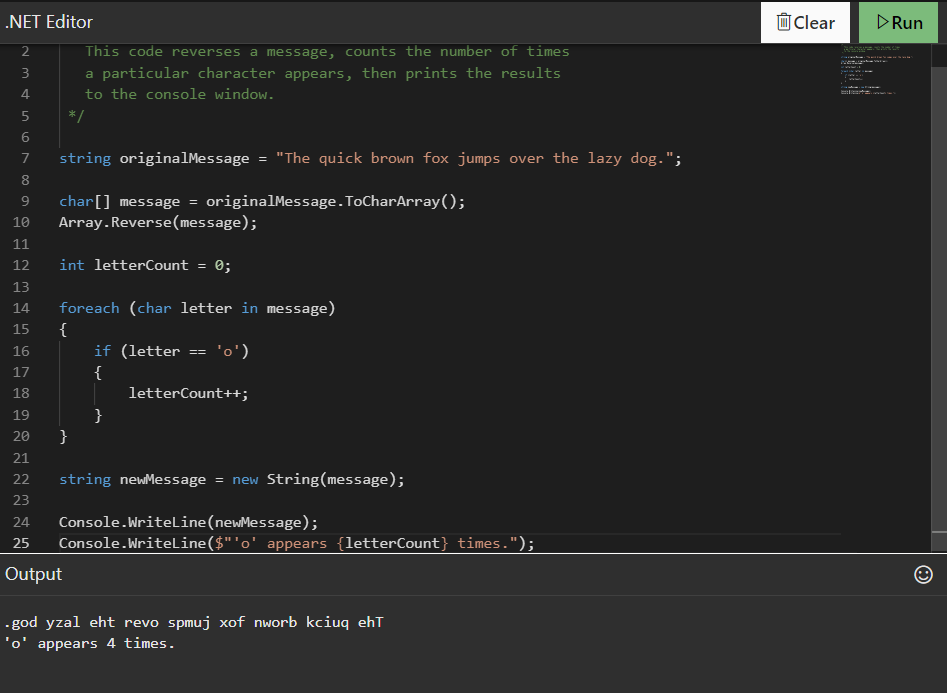
**Exercise - Comment your code**



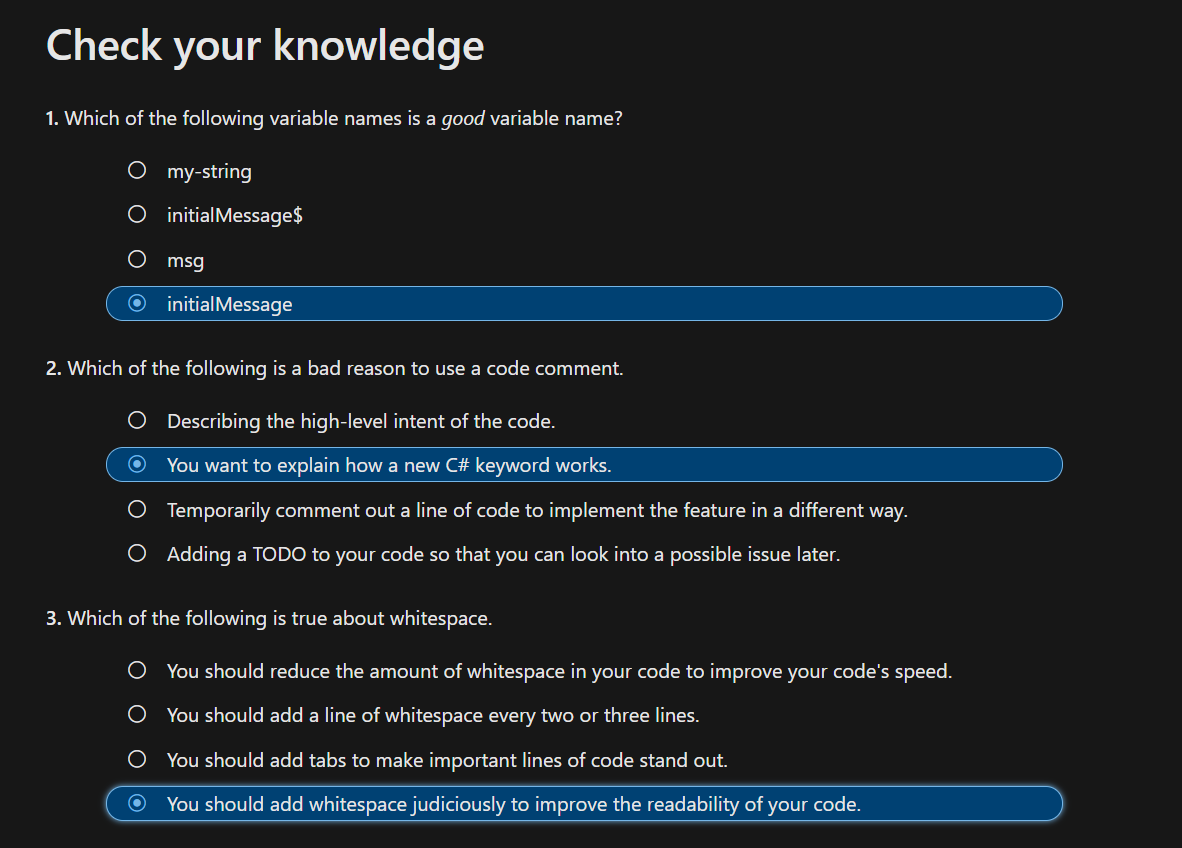
**Exercise - Use whitespace**



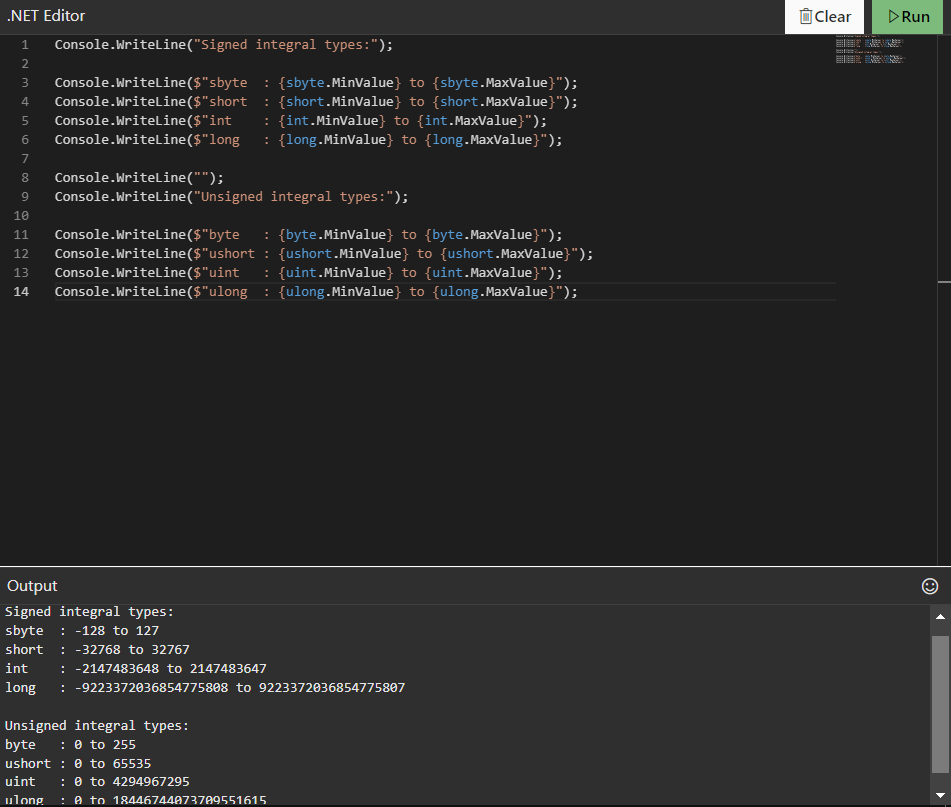
**Modify the code to make it more readable**



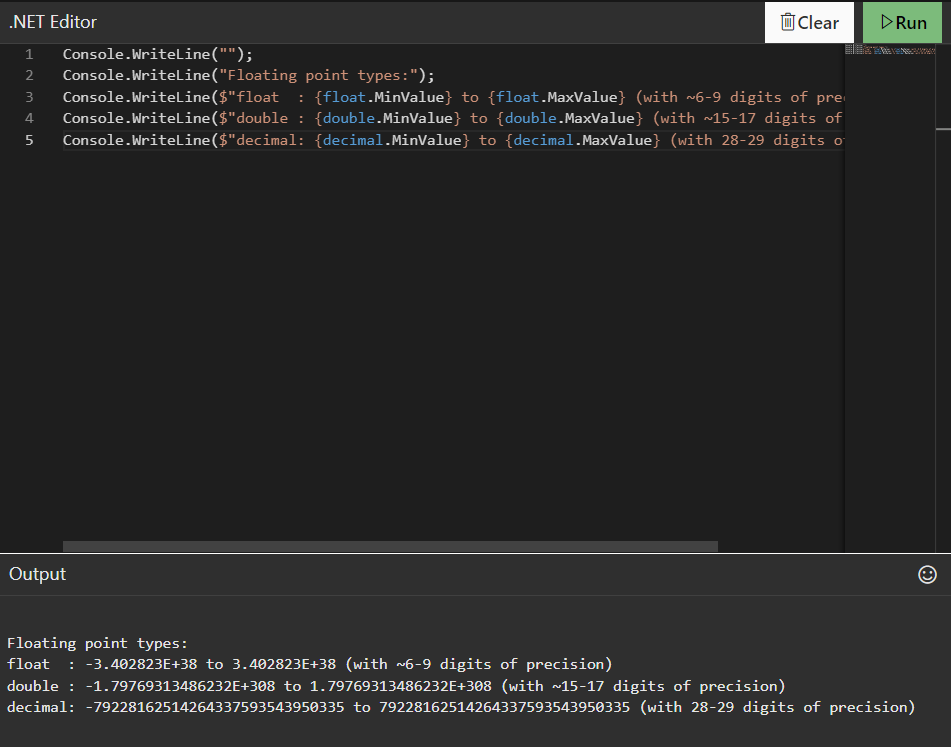
**MCQ :**



**Exercise - Integral types**



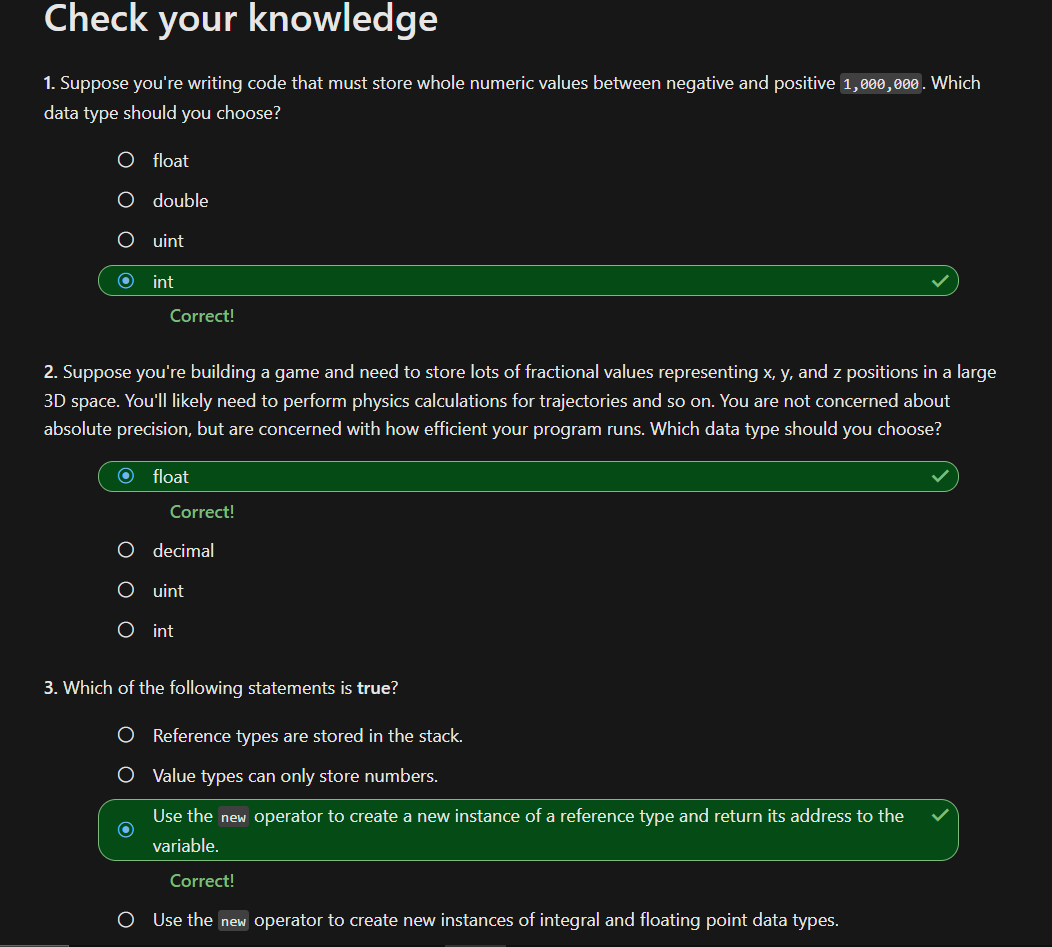
**Exercise - Floating-point types**



**Exercise - Reference types**



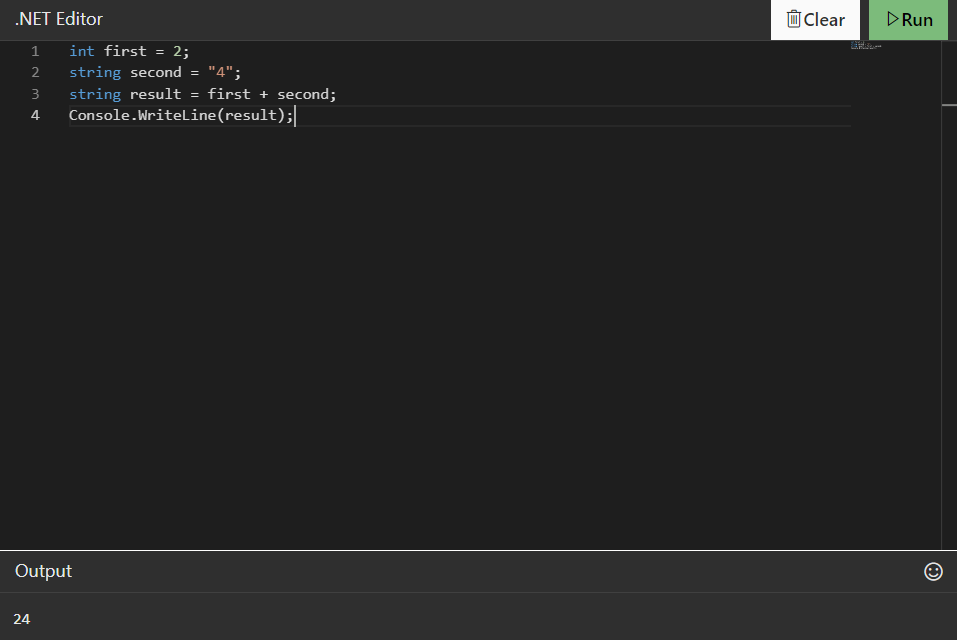
**MCQ :**



**SESSION -4 :**

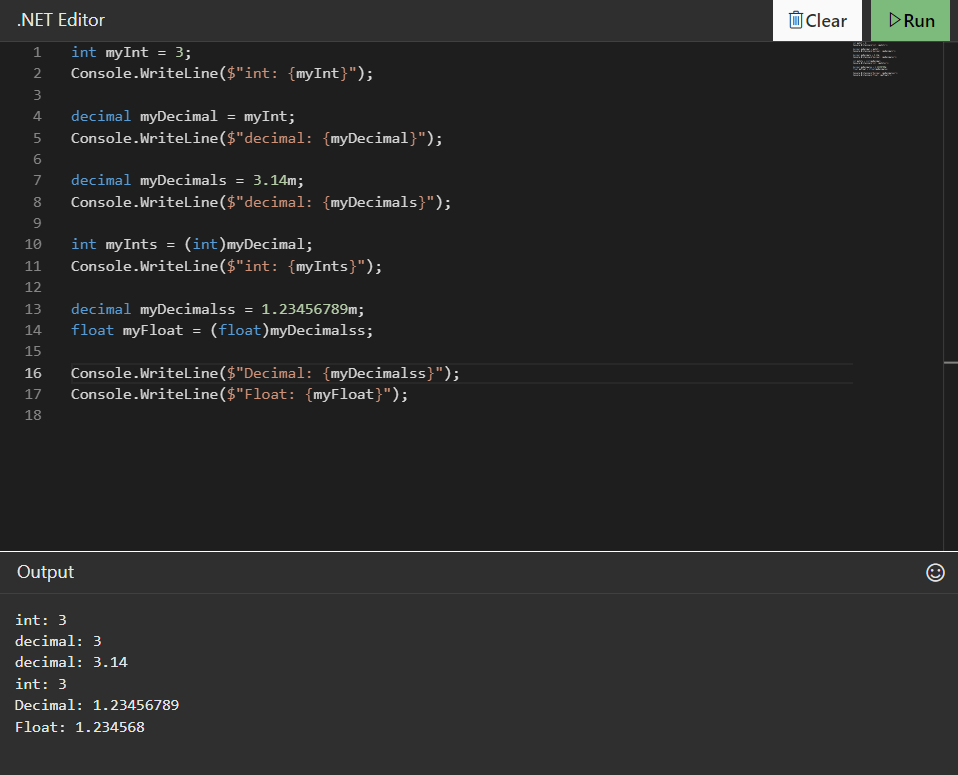
**Exercise - data type casting and conversion**

**Question 1: Is it possible, depending on the value, that attempting to change the value's data type would throw an exception at run time?**



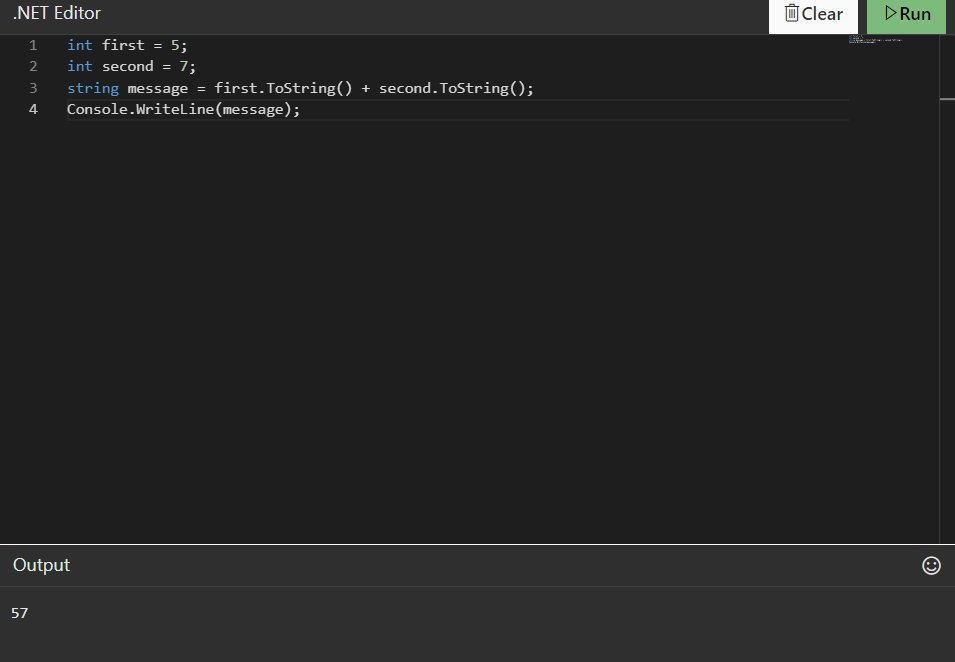
**Question 2: Is it possible, depending on the value, that attempting to change the value's data type would result in a loss of information?**

**Modify the code to perform an implicit conversion**



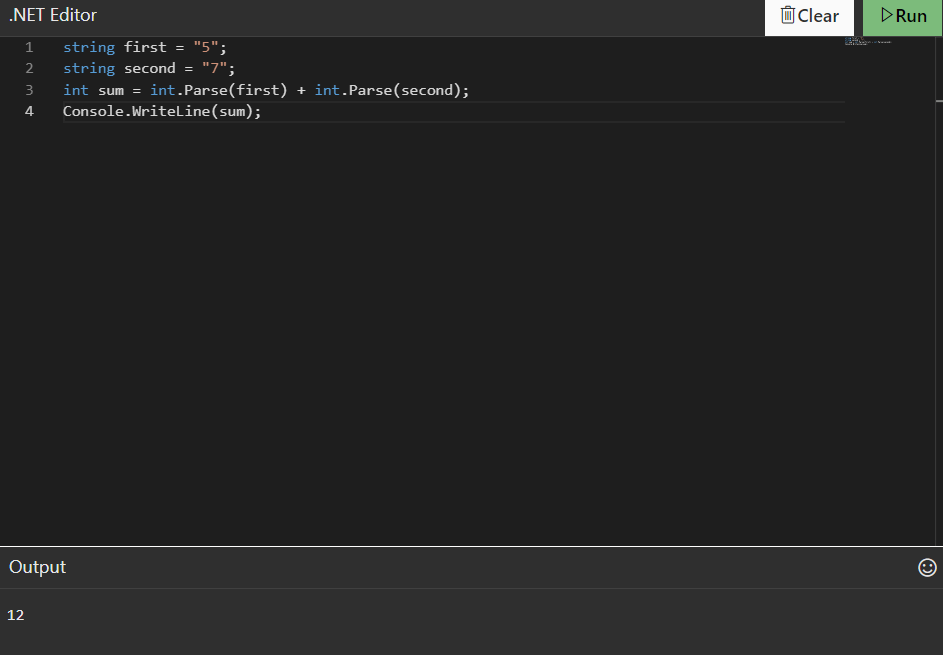
**Performing Data Conversions**

**Modify the code to convert a number to a string using the ToString() helper method**



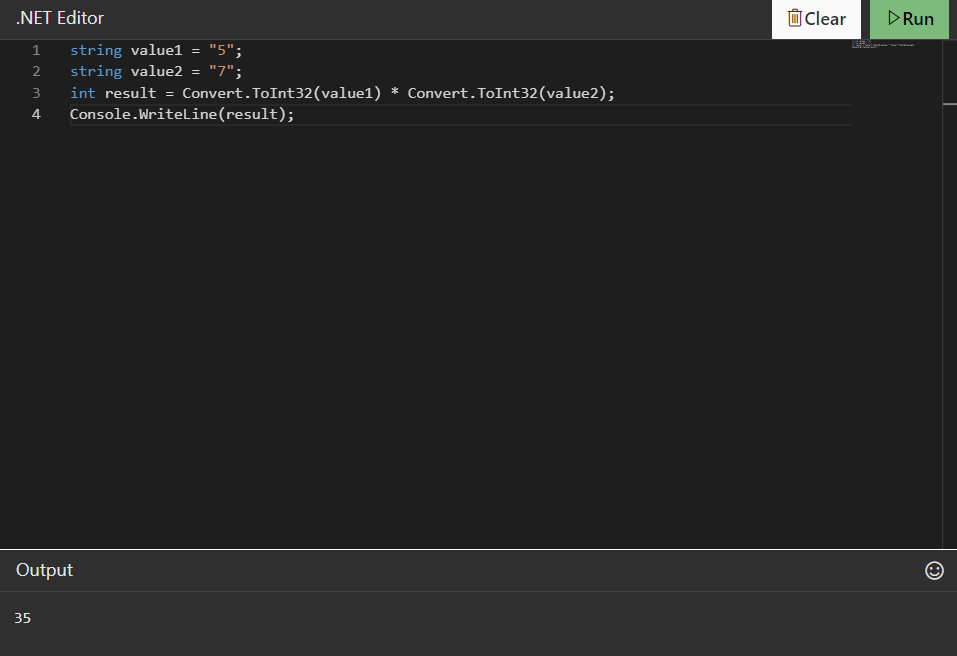
**Explicitly converting a string to a number**

**Modify the code to convert a string to an int using the Parse() helper method**

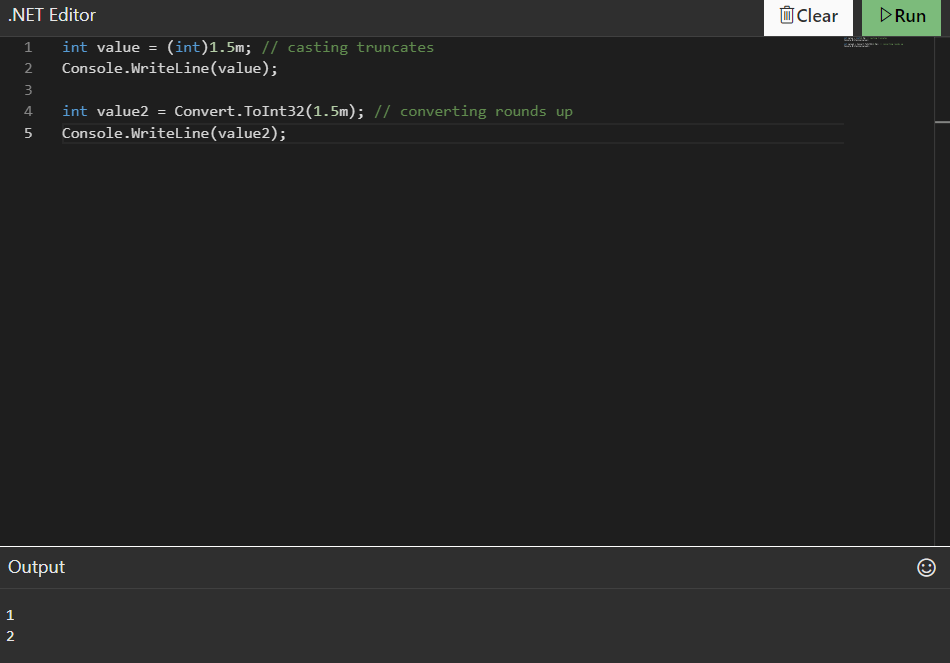


**Data Conversion using the Convert class**

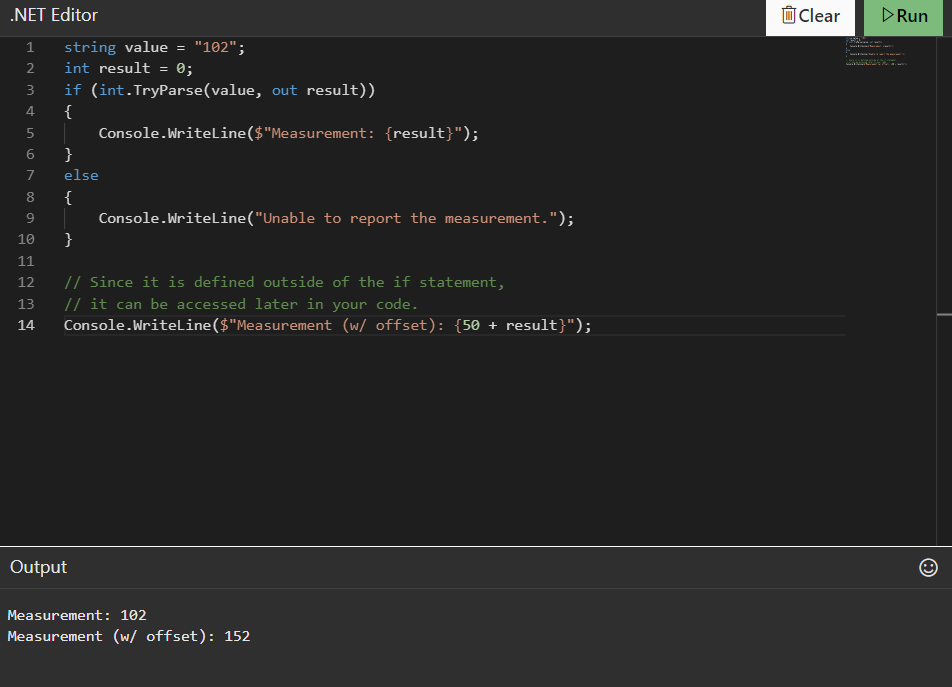
**Modify the code to convert a string to a number using the Convert class**



**Modify the code to compare casting and converting a decimal into an int**

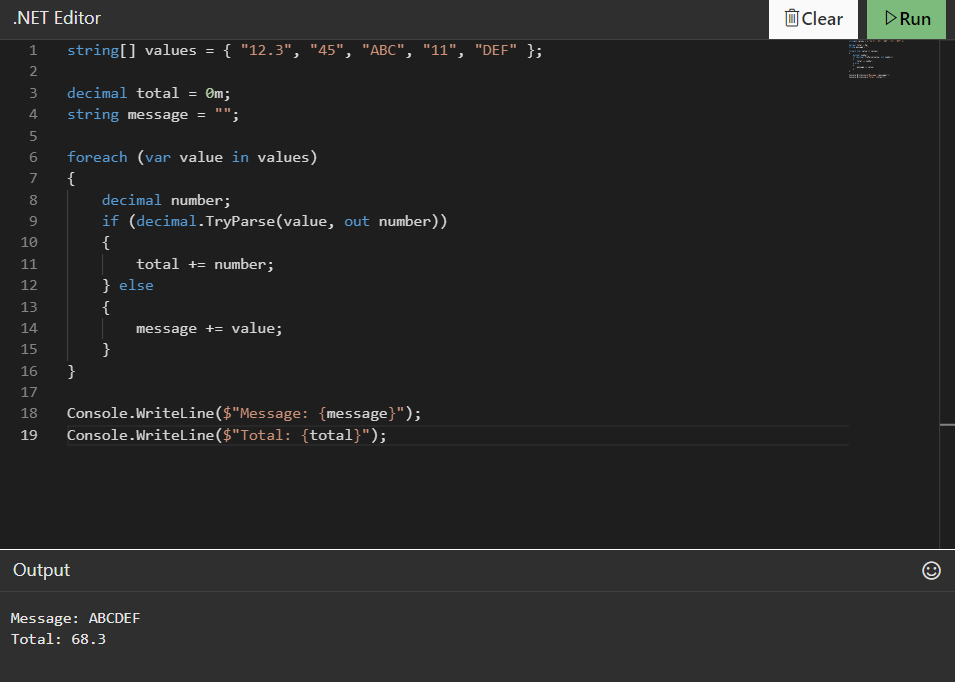


**Exercise - the TryParse() method**



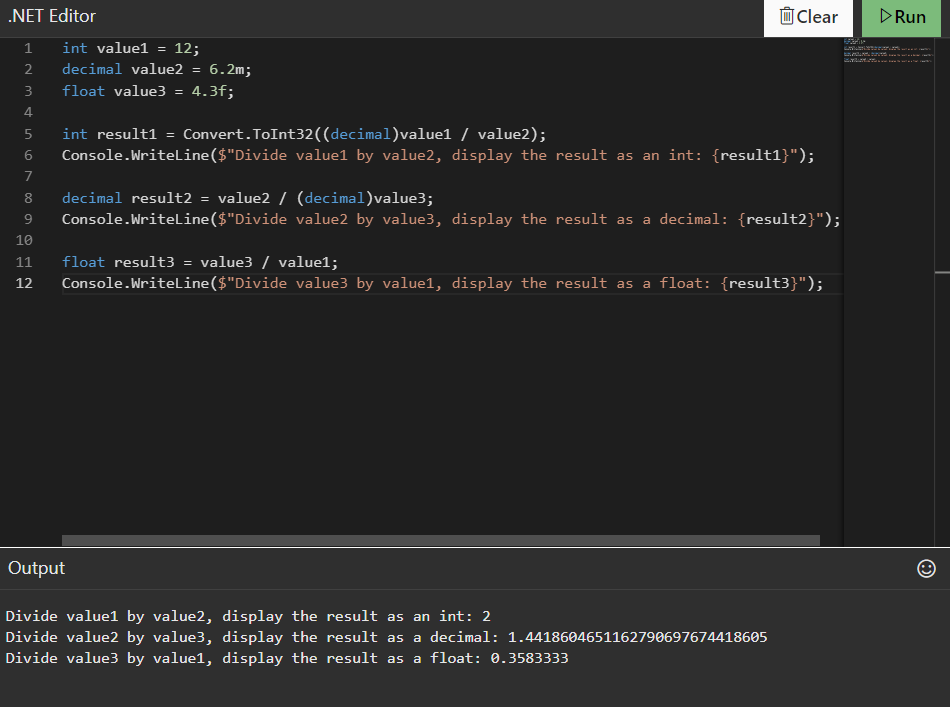
**First challenge**

**This module will feature two code challenges. This first challenge forces you to split up the data depending on its type and either concatenate or add the data accordingly.**

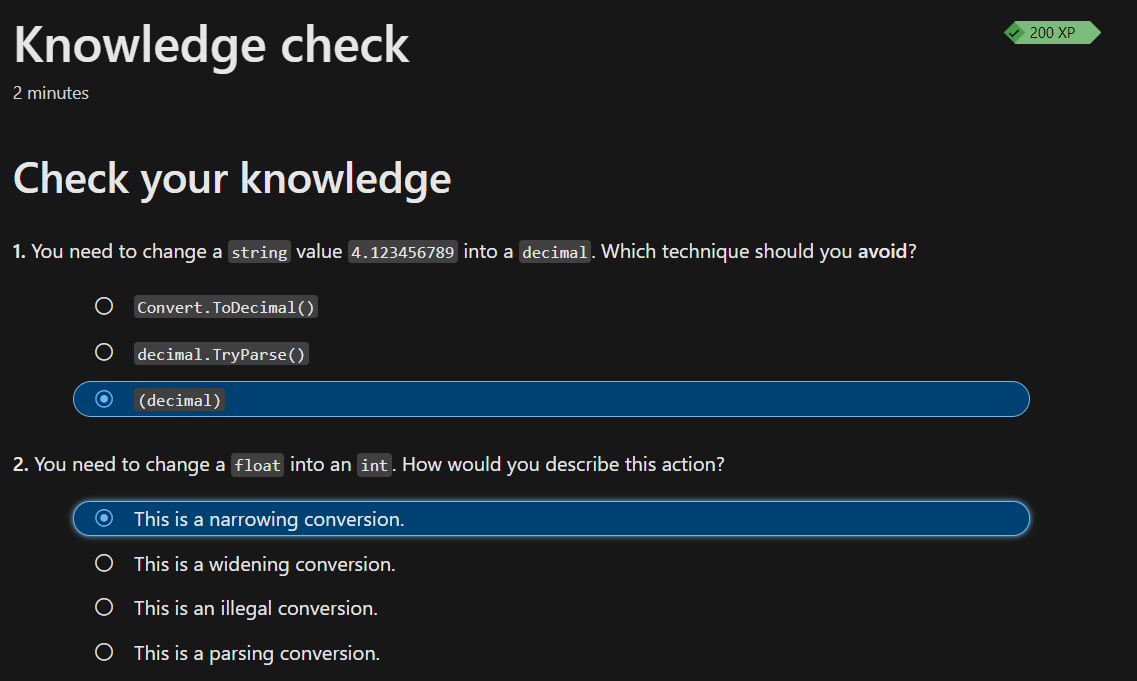


**Second challenge**

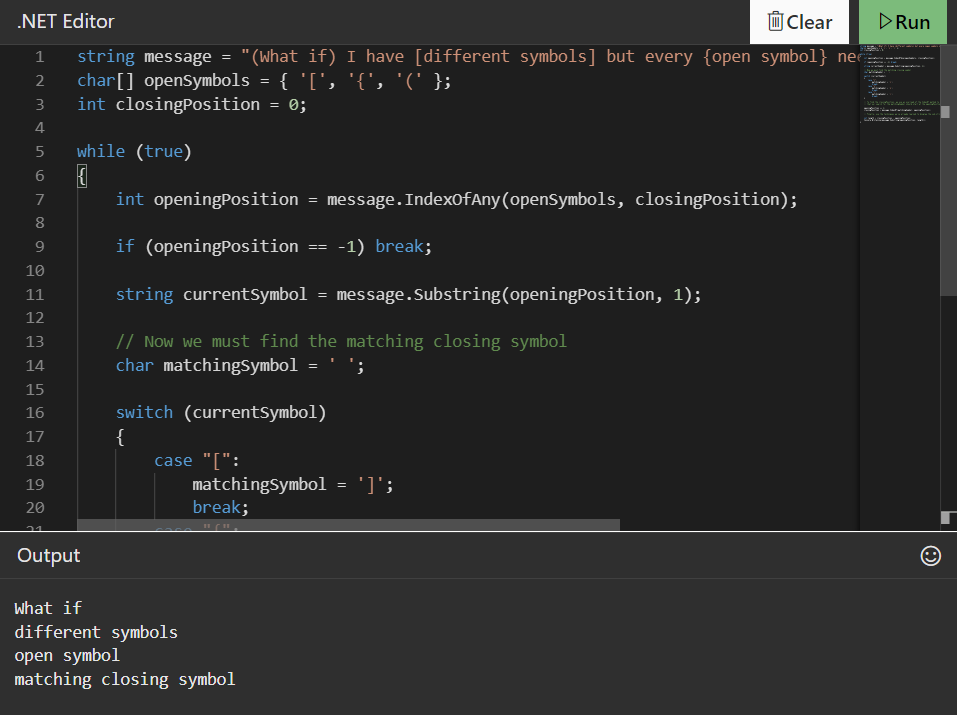
**The following challenge will force you to understand the implications of casting values considering the impact of narrowing and widening conversions.**



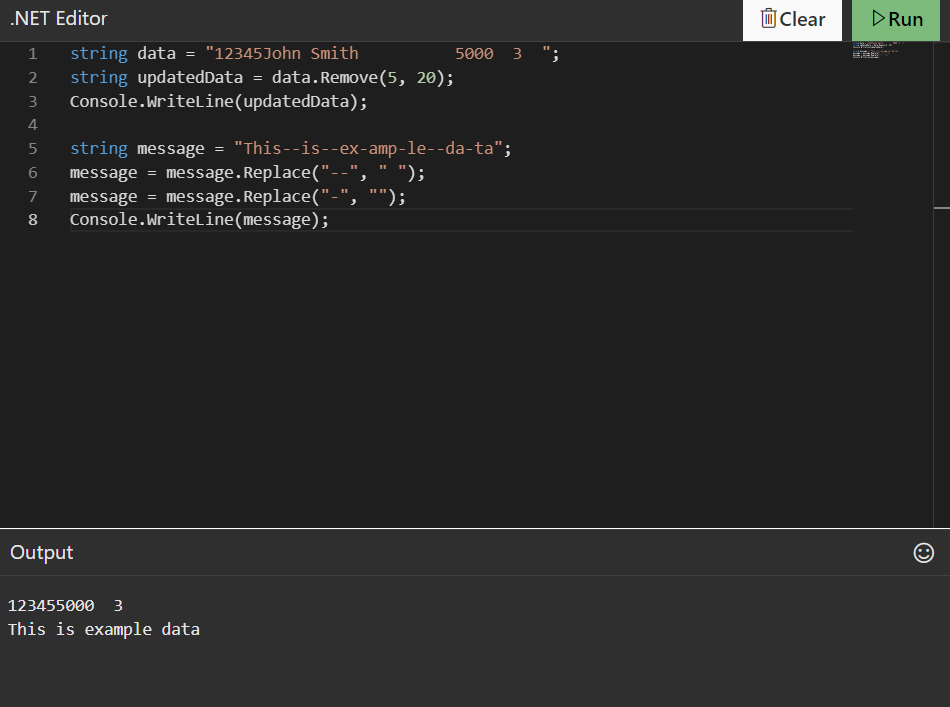
**MCQ :**



**Exercise - Use the string's IndexOf() and Substring() helper methods**

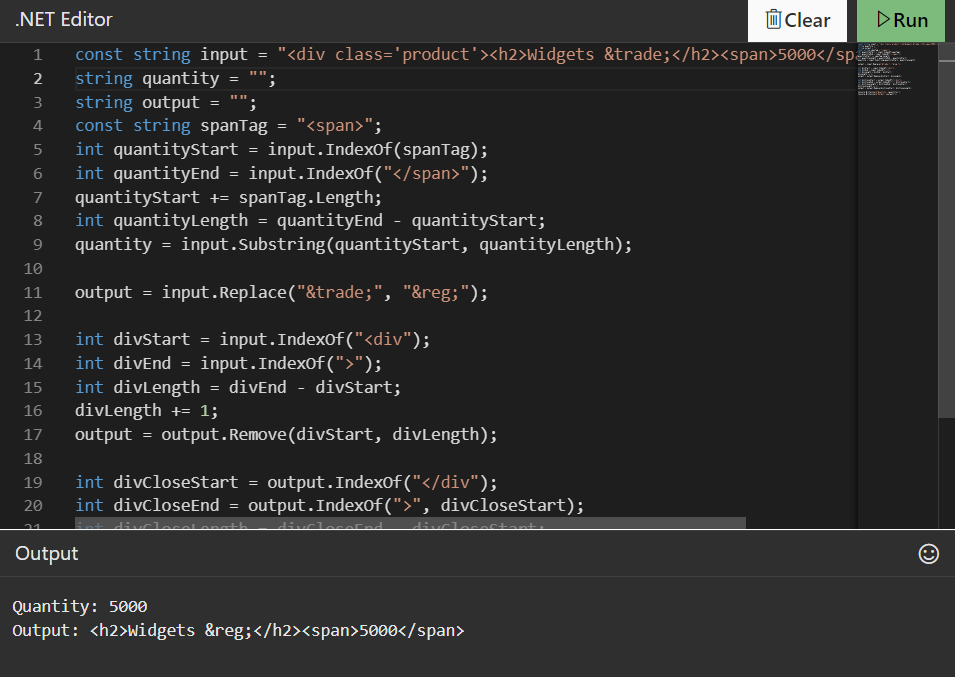


**Exercise - Use the Remove() and Replace() methods**

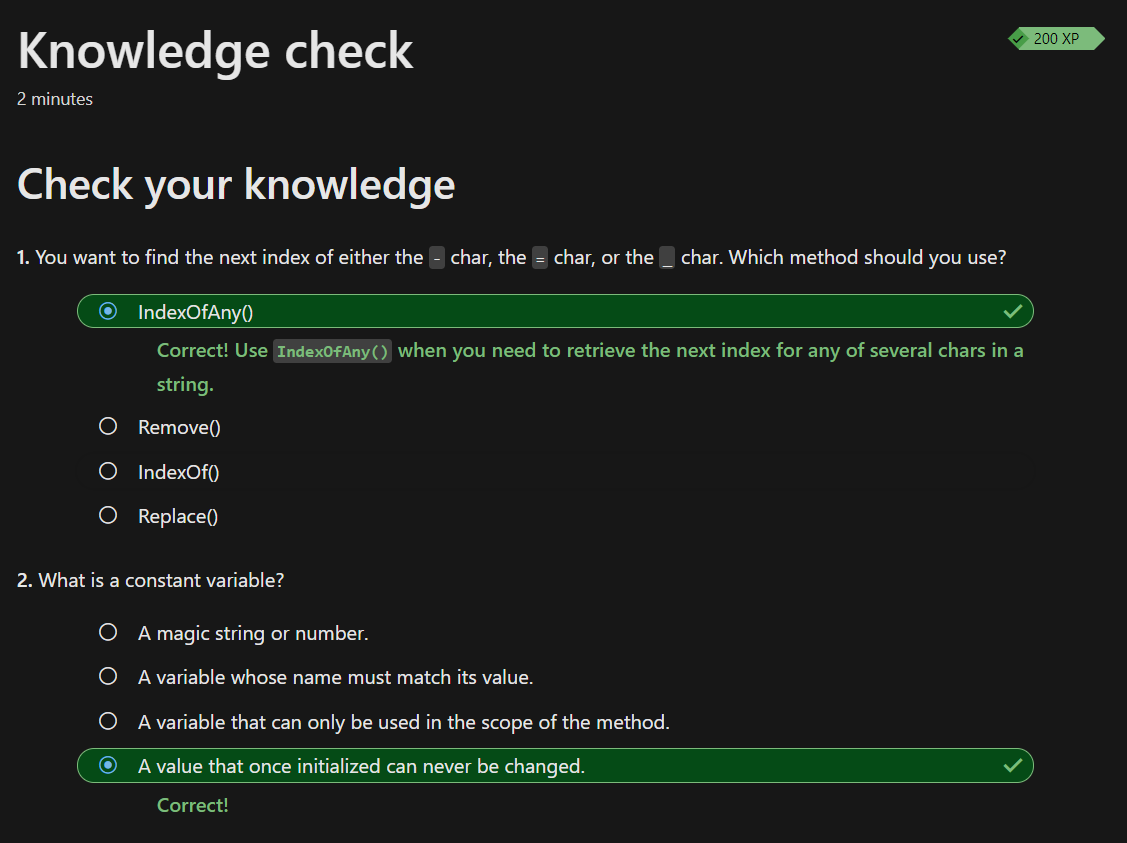


**Challenge**

**In this challenge, you'll work with a string that contains a fragment of HTML. You'll extract data from the HTML fragment, replace some of its content, and remove other parts of its content to achieve the desired output.**

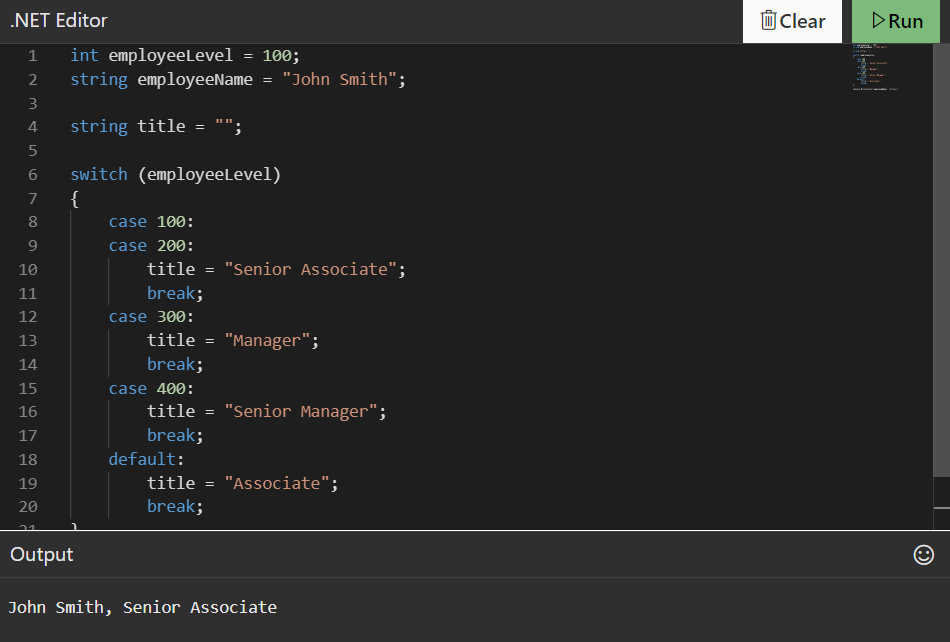


**MCQ :**



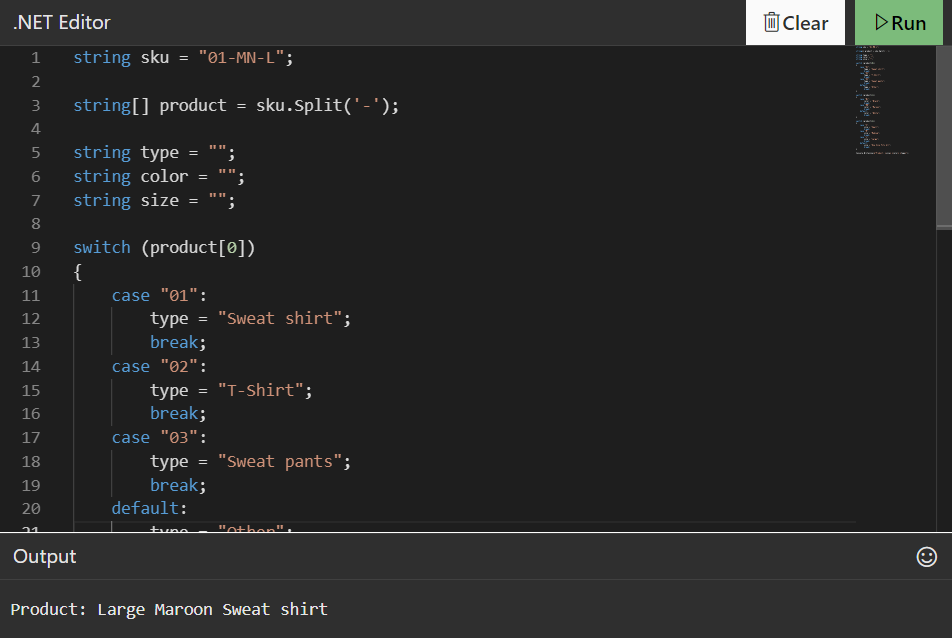
**SESSION -5 :**

**Exercise - Use the switch case statement**

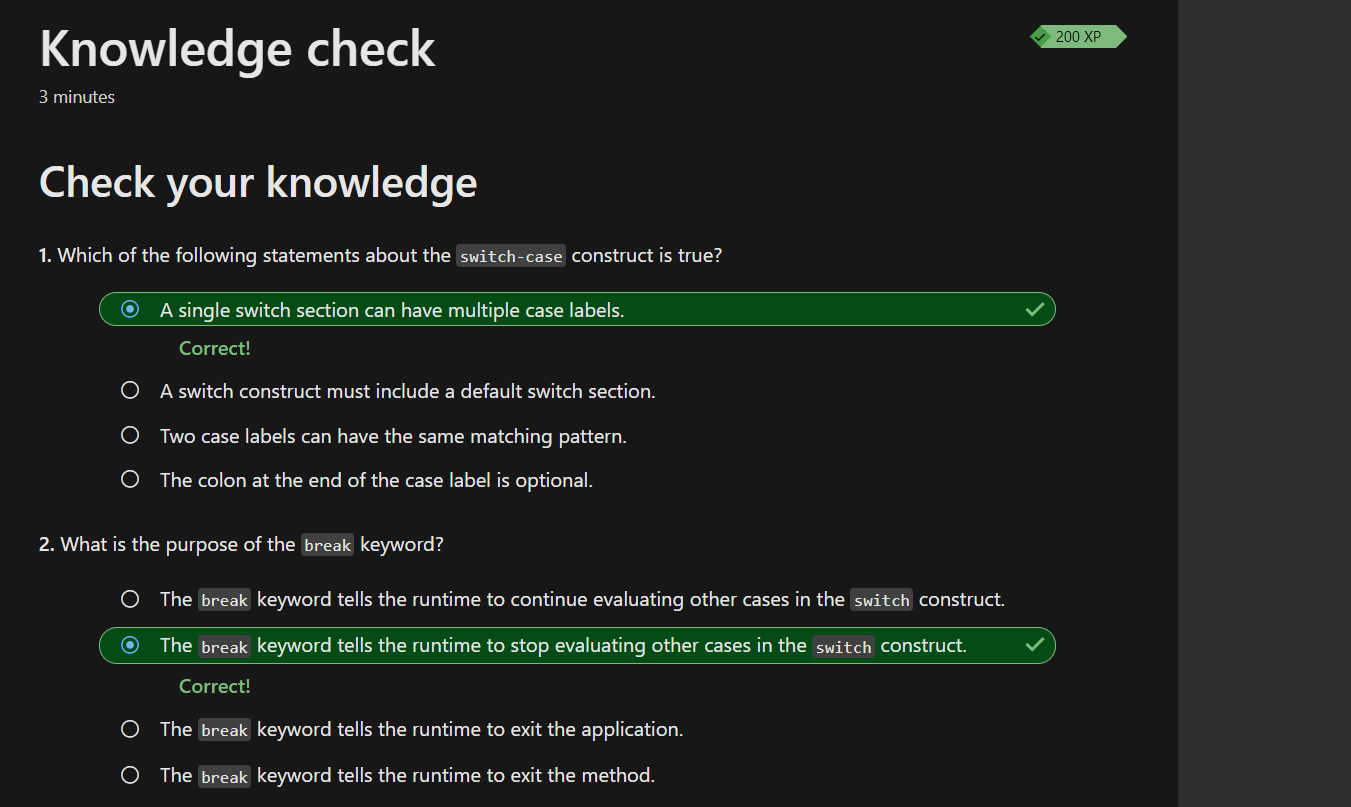


**Challenge**

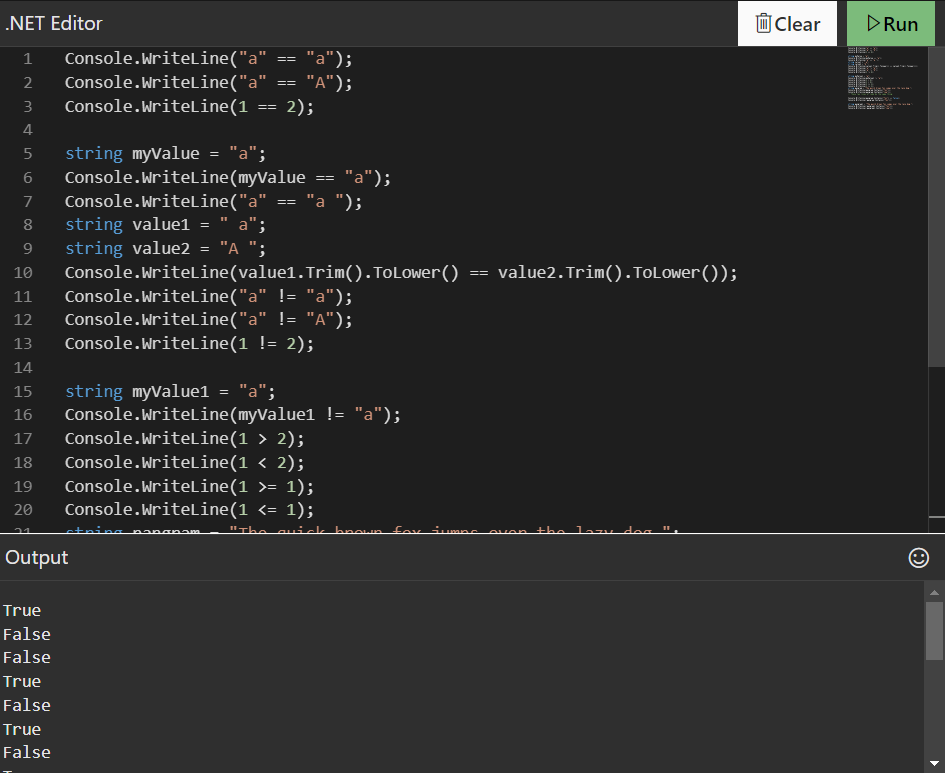
**Look up product by SKU challenge**



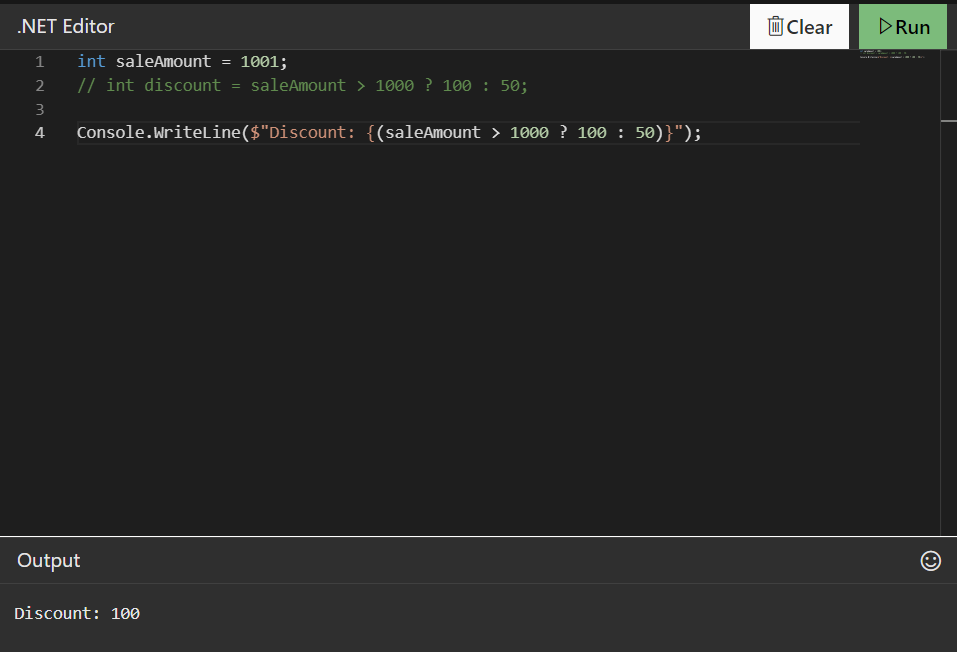
**MCQ :**



**Exercise - Boolean Expressions**



**Exercise - conditional operator**



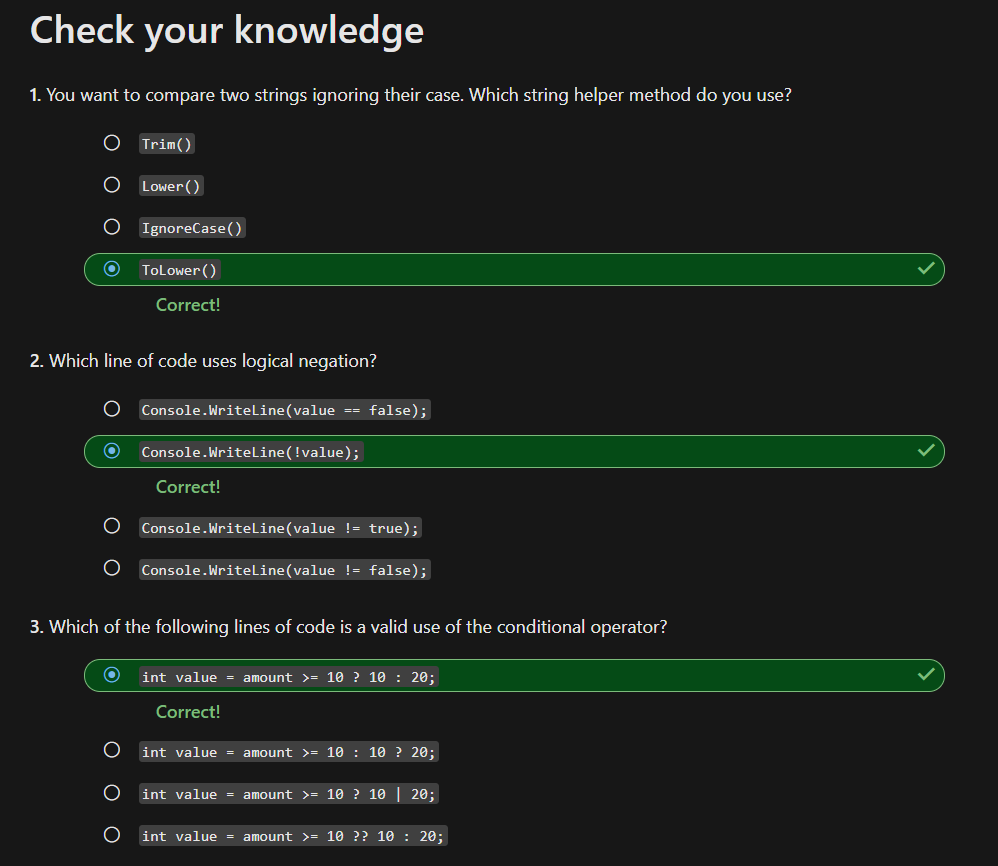
**Heads or tails challenge**



**Complicated Permissions Challenge**

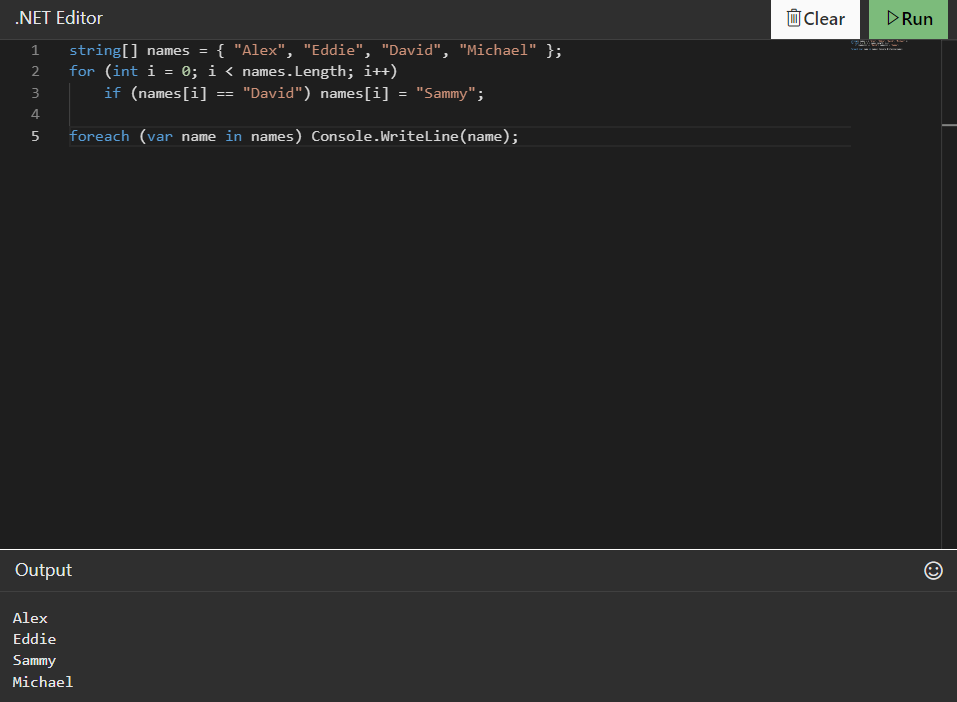


**MCQ :**

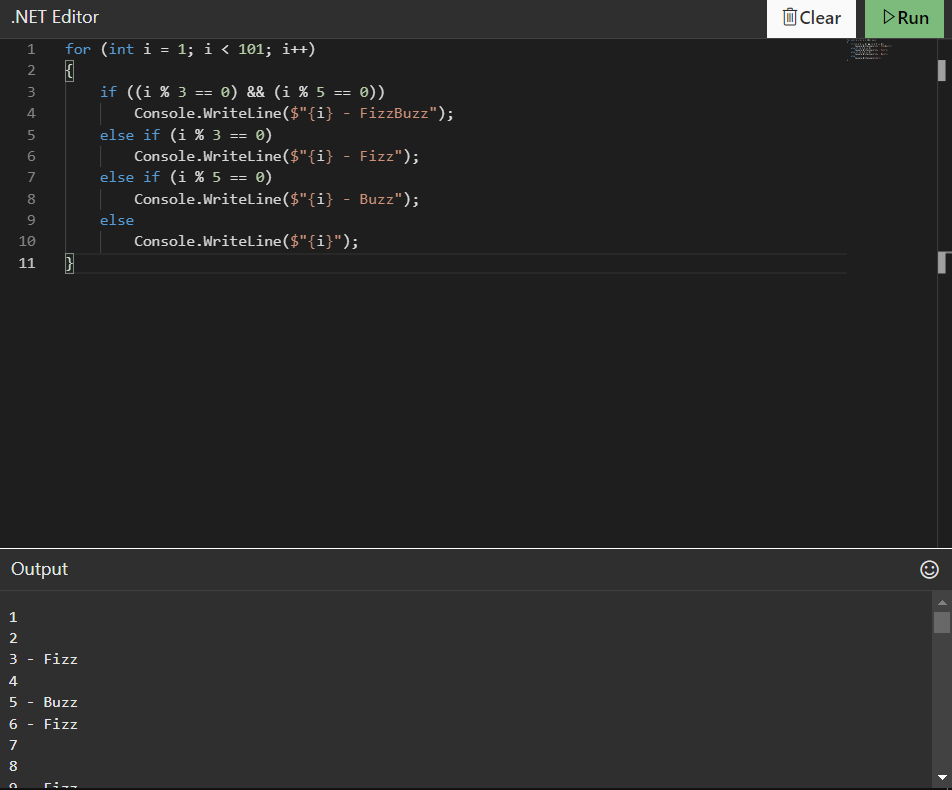


**SESSION -6 :**

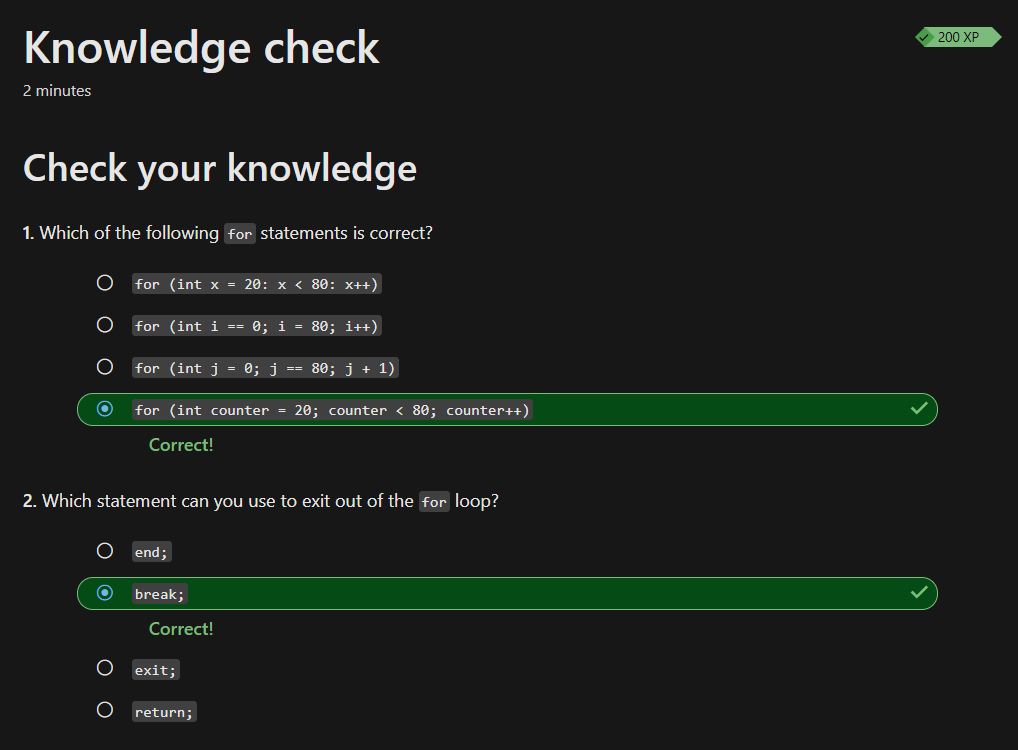
**Exercise - for iteration statement**



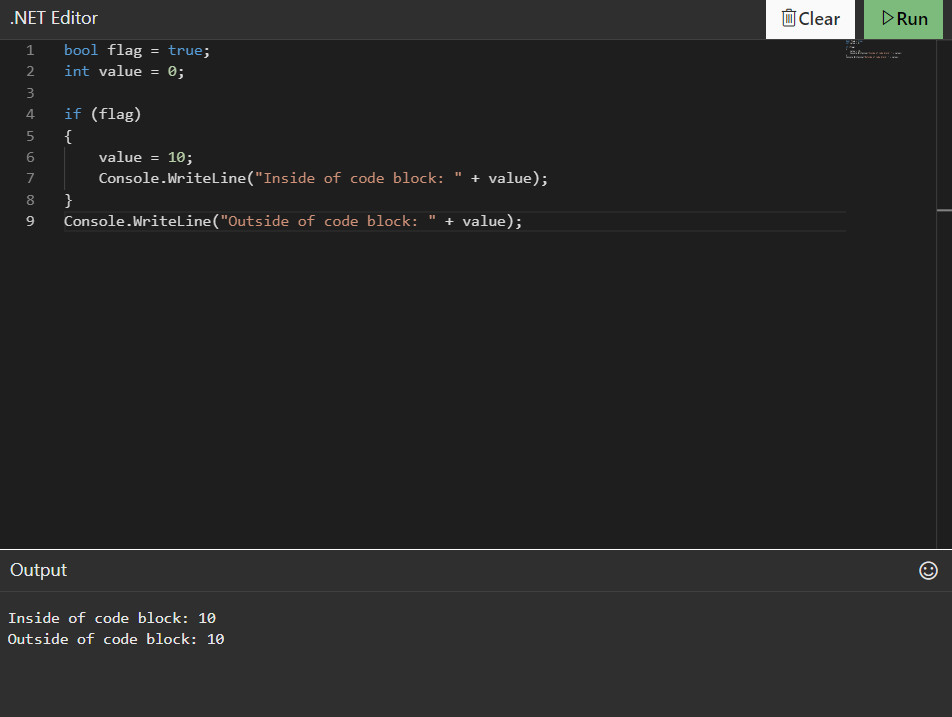
**FizzBuzz challenge**



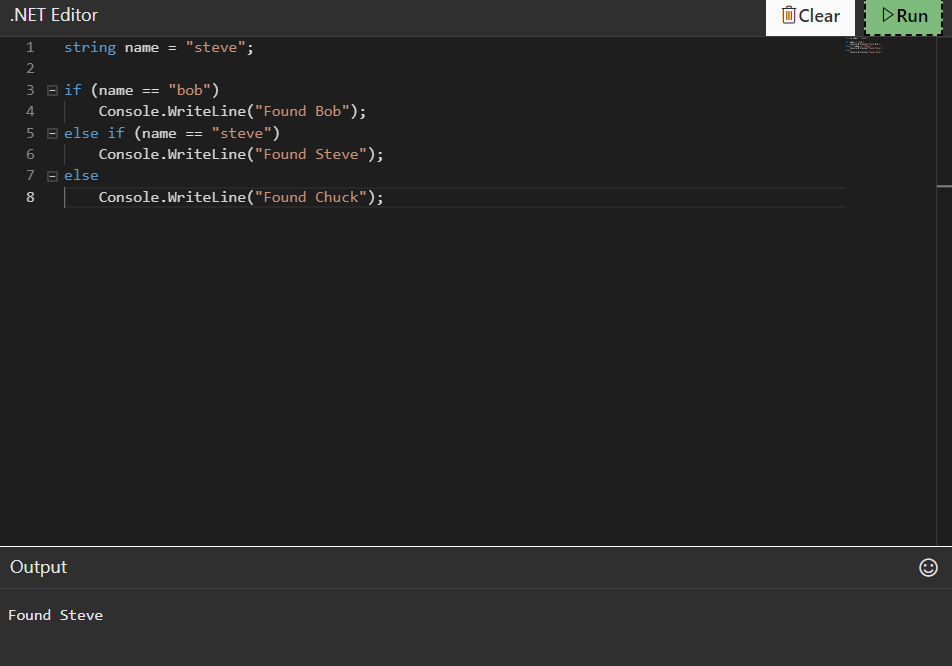
**MCQ :**



**Exercise - Code blocks and variable scope**



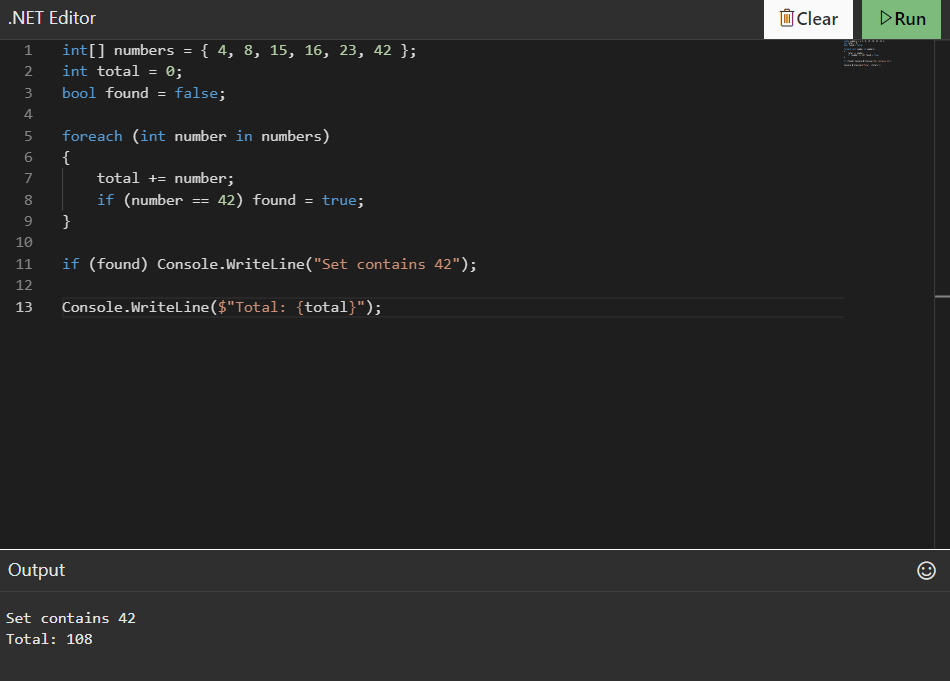
**Exercise - Remove code blocks in if statements**



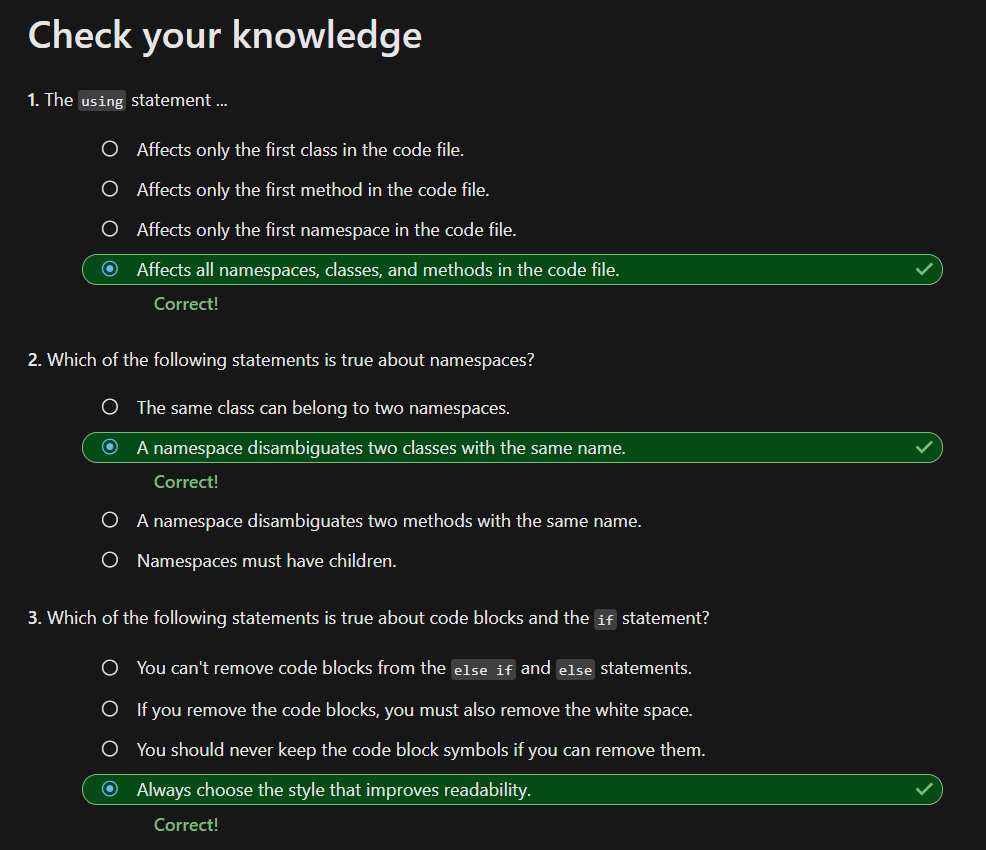
**Challenge**

**Fix this code**

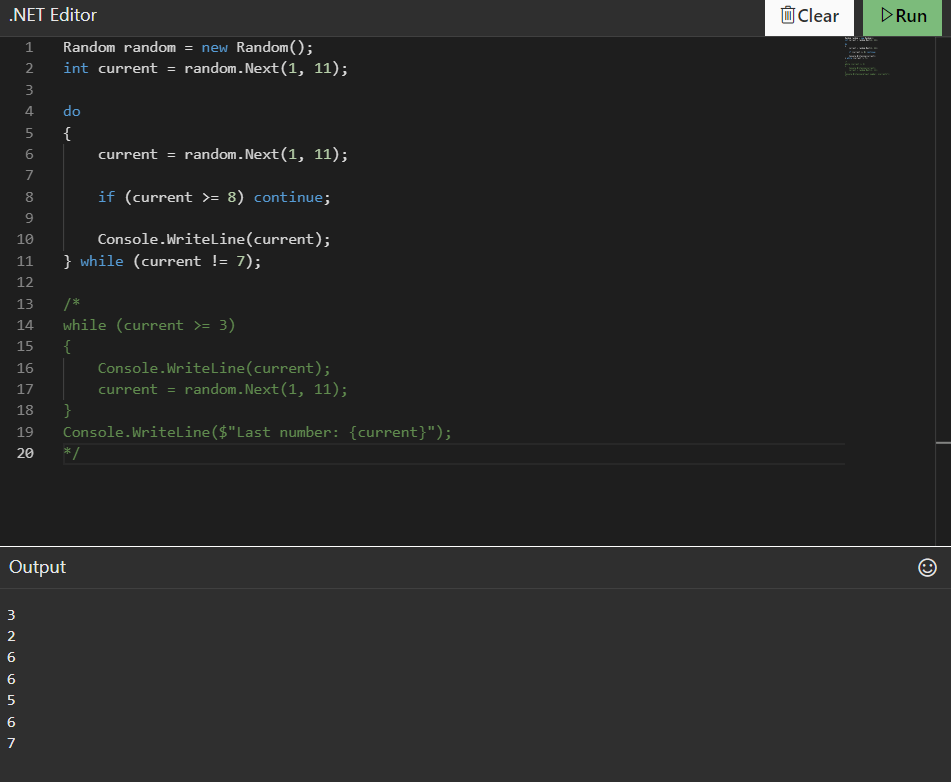
**Use what you've learned in this module to fix this poorly written code. There are many improvements that you can make.**



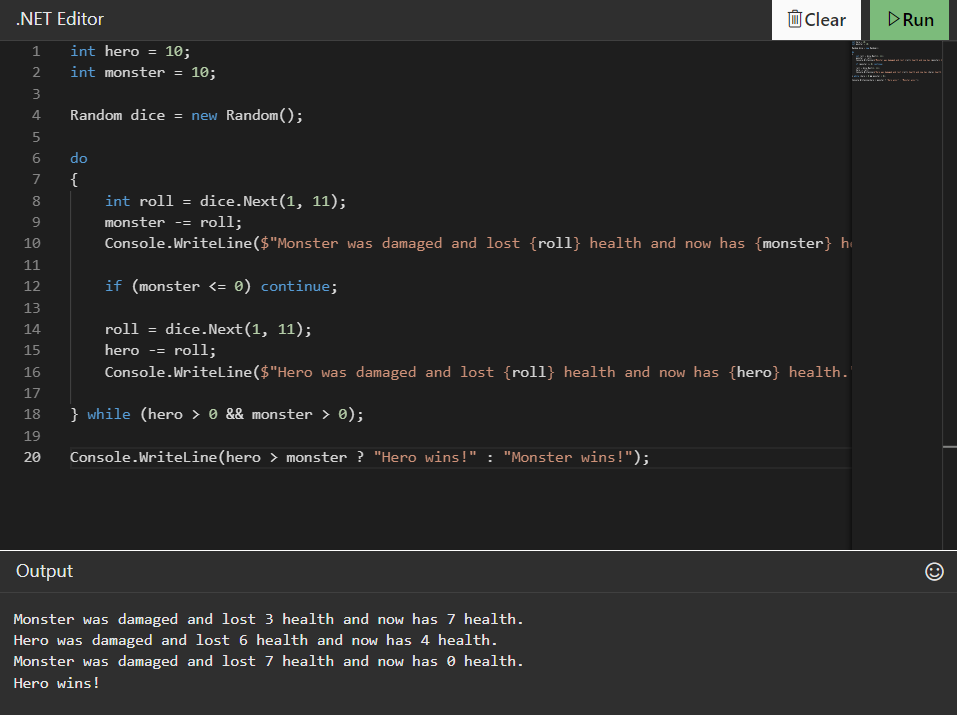
**MCQ :**



**Exercise - do-while, while, and continue statements**



**Role playing game battle challenge**



**MCQ :**

